

# SISTEM KLASIFIKASI JENIS TANAMAN HIAS DAUN PHILODENDRON MENGGUNAKAN METODE *K-NEAREST NEIGHBOOR* (KNN) BERDASARKAN NILAI *HUE, SATURATION, VALUE* (HSV)

Dani Syahid<sup>1</sup>, Jumadi<sup>2</sup>, Dian Nursantika<sup>3</sup>

<sup>1,2,3</sup>Jurusan Teknik Informatika, Fakultas Sains dan Teknologi

Universitas Islam Negeri Sunan Gunung Djati Bandung

Jl. A.H. Nasution 105, Bandung 40614 Indonesia

<sup>1</sup>danisyahid10@gmail.com, <sup>2</sup>jumadi@uinsgd.ac.id, <sup>3</sup>dianursantika@yahoo.com

**Abstrak** --Tanaman hias daun biasa digunakan untuk mempercantik halaman pekarangan rumah dengan aneka warna yang indah pada tanaman hias daun ini menjadi bahan perhatian khususnya bagi pecinta tanaman. Namun dengan banyaknya jenis tanaman hias membuat kita sulit untuk mengetahui nama tumbuhan yang kita minati. Sistem pendeteksi citra tanaman hias daun bekerja dengan cara membandingkan data citra latih yang telah tersimpan pada database terhadap data citra yang akan diuji. Data citra uji akan diklasifikasikan dengan menggunakan penerapan metode *K-Nearest Neighbor* yaitu berfungsi untuk menghitung jarak terdekat antara data citra latih dan data citra uji pada setiap pikselnya. Setiap piksel pada citra akan dilakukan proses konversi *red, Green, Blue* (RGB) ke dalam ekstraksi fitur warna *hue, saturation, value* (HSV) terlebih dahulu. Setelah didapat nilai HSV, maka dilakukan proses klasifikasi menggunakan metode KNN. Data sampel pada penelitian ini menggunakan 5 klasifikasi citra data latih dengan 10 data citra uji pada setiap data citra latih. Pada penelitian ini, diperoleh hasil dari akurasi sistem pendeteksi citra tanaman dengan hasil mencapai 92%.

**Kata kunci** -- *Hue, Saturation, Value* (HSV), *K-Nearest Neighbor* (KNN), Tanaman hias daun *Philodendron*.

## I. PENDAHULUAN

Model HSV mendefinisikan warna dalam terminologi *hue, saturation* dan *value*. *Hue* menyatakan warna sebenarnya, seperti merah, violet, dan kuning. *Hue* digunakan untuk membedakan warna-warna dan menentukan kemerahan (*redness*), Kehijauan (*greenness*) dari cahaya [2].

*K-Nearest Neighbor* (K-NN) merupakan metode yang dapat digunakan untuk proses klasifikasi atau mengelompokan data citra uji yang sudah ditentukan kelasnya pada data latih yang sudah disimpan pada *datastore*[1]. Proses pengelompokan ini akan dilakukan dari hasil perhitungan HSV yang sudah dilakukan pada data latih sebelumnya. Untuk menghitung jarak kemiripan fitur warna antara citra

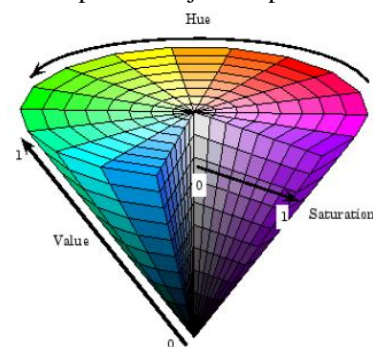
data latih dan data citra yang akan diuji dibutuhkan metode yang dapat melakukan perhitungan jarak kemiripan antara citra latih dan citra uji.

## II. LANDASAN TEORI

### A. Model Warna *Hue, Saturation, Value* (HSV)

Model warna HSV mendefinisikan warna dalam terminologi *Hue, Saturation*, dan *Value*. *Hue* menyatakan warna sebenarnya, seperti merah, violet, dan kuning. *Hue* digunakan untuk membedakan warna-warna dan menentukan kemerahan (*redness*), Kehijauan (*greenness*) dari cahaya [2].

Nilai *hue* antara 0 sampai 1 berarti warna antara merah melewati kuning, hijau, cyan, biru dan magenta dan kembali menjadi merah. Nilai *saturation* antara 0 sampai 1 berarti dari tidak tersaturasi (keabuan) sampai tersaturasi penuh (tidak putih) [3]. Nilai *value* atau *brightness* antara 0 sampai 1 berarti warna semakin cerah seperti ditunjukkan pada Gambar 1.



Gambar 1. Model Warna HSV <sup>1</sup>

Nilai HSV yang dijadikan tolak ukur dalam penelitian ini dilakukan tahap normalisasi kedalam angka yang lebih sederhana. Hal ini bertujuan untuk mempersingkat waktu komputasi, namun tidak mengurangi akurasi pencarian secara signifikan [3].

<sup>1</sup>Gonzales, R.C., and Woods, R.E. 2002. *Digital Image Processing* Second edition. New Jersey: Prentice Hall.

Quantisasi dilakukan dengan persamaan dibawah ini[4]:

$$H \begin{cases} 0 \text{ if } h \in [316,20] \\ 1 \text{ if } h \in [21,40] \\ 2 \text{ if } h \in [41,75] \\ 3 \text{ if } h \in [76,155] \\ 4 \text{ if } h \in [156,190] \\ 5 \text{ if } h \in [191,270] \\ 6 \text{ if } h \in [271,295] \\ 7 \text{ if } h \in [296,315] \end{cases} \quad S \begin{cases} 0 \text{ if } s \in [0.0,2] \\ 1 \text{ if } s \in [0.2,0.7] \\ 2 \text{ if } s \in [0.7,1] \end{cases} \quad V \begin{cases} 0 \text{ if } v \in [0.0,0.2] \\ 1 \text{ if } v \in [0.2,0.7] \\ 2 \text{ if } v \in [0.7,1] \end{cases}$$

(Persamaan 1) [4].

Vektor 3 dimensi HSV tersebut dikonversi menjadi vector 1 dimensi dengan tetap mempertimbangkan bobot dari masing – masing nilai komponen HSV[4]. Tujuannya agar nilai hsv ini dapat dicari jaraknya melalui *Euclidean Distance*. Persamaan yang digunakan oleh Kavitha [4] adalah:

$$G = (9 * H) + (3 * S) + V$$

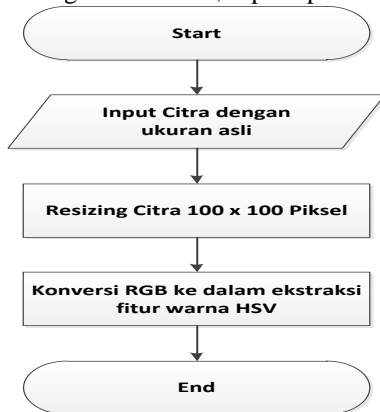
(Persamaan 2)[4].

### III. METODE PENELITIAN

Data yang digunakan dalam penelitian ini yaitu citra berwarna pada proses pendeteksian citra yaitu dengan melakukan proses ekstraksi citra warna.

#### A. Metode perhitungan Ekstraksi warna HSV.

Adapun gambaran flow chart (diagram alir) untuk proses perhitungan fitur HSV, seperti pada Gambar II.



Gambar II. flow chart perhitungan fitur HSV.

#### B. Metode K-Nearest Neighbor(KNN)

Pengertian klasifikasi yaitu proses pengelompokan, artinya mengumpulkan benda / entitas yang sama serta memisahkan benda/entitas yang tidak sama[1]. KNN (*K-Nearest Neighbor*) adalah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat atau memiliki persamaan ciri paling banyak dengan objek tersebut. Dekat atau jauhnya tetangga biasanya dihitung dengan jarak *Euclidean*. Teknik ini

sederhana dan dapat memberikan akurasi yang baik terhadap hasil klasifikasi[1].

$$d(i,j) = \sqrt{|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \dots + |x_{ip} - x_{jp}|^2}$$

(Persamaan 3)

Keterangan :  $d(i,j)$  = nilai jarak  
 $x_i$  = nilai – nilai pada fitur 1  
 $x_j$  = nilai – nilai pada fitur 2

(Persamaan 3)

Dalam algoritma ini, nilai k yang terbaik itu tergantung pada jumlah data. Ukuran nilai k yang besar belum tentu menjadi nilai k yang terbaik begitu juga sebaliknya[1].

Langkah – langkah untuk menghitung algoritma KNN :

1. Menentukan nilai k.
2. Menghitung kuadrat jarak *euclid (query instance)* masing – masing objek terhadap *training data*.
3. Mengurutkan objek – objek tersebut ke dalam kelompok yang mempunyai jarak *euclid* terkecil.
4. Mengumpulkan label *class Y* (klasifikasi *Nearest Neighborhood*).

Perhitungan diatas menggunakan persamaan (3) pada bab sebelumnya, dapat dibuatkan sebuah flow chart (diagram alir) seperti pada Gambar III.



Gambar III flow chart proses perhitungan KNN

#### C. Analisis Proses Perhitungan Nilai HSV

Sebagai contoh, terdapat suatu citra berukuran 5 x 5 piksel memiliki nilai (H, S, V). Nilai *Hue*, *Saturation* dan *Value* tersebut dikuantisasi menggunakan persamaan (1) pada bab II[4].

H : 25, S : 0.4,	H : 5, S : 0.8,	H : 200, S : 1.0,	H : 22, S : 0.8,	H : 125, S : 0.0,
---------------------	--------------------	----------------------	---------------------	----------------------

V : 0.5	V : 0.0	V : 0.7	V : 0.3	V : 0.2
H : 133, S : 0.0, V : 0.5	H : 215, S : 0.6, V : 0.0	H : 195, S : 0.2, V : 0.2	H : 16, S : 0.4, V : 0.2	H : 73, S : 0.1, V : 0.1
H : 259, S : 0.1, V : 0.5	H : 231, S : 0.1, V : 0.4	H : 163, S : 0.0, V : 0.2	H : 11, S : 0.2, V : 0.5	H : 212, S : 0.3, V : 0.3
H : 357, S : 0.6, V : 0.6	H : 277, S : 0.2, V : 0.2	H : 43, S : 0.2, V : 0.5	H : 174, S : 0.1, V : 0.0	H : 93, S : 0.8, V : 1.0
H : 139, S : 0.3, V : 0.5	H : 120, S : 0.3, V : 0.8	H : 31, S : 0.1, V : 0.2	H : 222, S : 0.2, V : 0.8	H : 320, S : 0.8, V : 0.5

**Gambar IV** Ilustrasi Nilai HSV tiap piksel pada sebuah Citra

Kemudian untuk untuk mengkombinasikan ketiga nilai HSV tersebut menjadi satuan nilai warna menggunakan persamaan (2) [4].

H 1	H 5	H 5	H 1	H 3
S 1	S 2	S 2	S 2	S 0
V 1	V 0	V 1	V 1	V 0
H 3	H 5	H 5	H 0	H 2
S 0	S 1	S 0	S 1	S 0
V 1	V 0	V 0	V 0	V 0
H 5	H 5	H 4	H 0	H 5
S 0	S 0	S 0	S 0	S 1
V 1	V 1	V 0	V 1	V 1
H 0	H 6	H 2	H 4	H 3
S 1	S 0	S 0	S 0	S 2
V 1	V 0	V 1	V 0	V 2
H 3	H 3	H 1	H 5	H 0
S 1	S 1	S 0	S 0	S 2
V 1	V 2	V 0	V 2	V 1

**Gambar V.** Ilustrasi Kuantisasi Nilai HSV pada setiap piksel citra

Kumulatif dari perhitungan seluruh piksel pada citra tersebut menghasilkan 1 dimensi vektor, vektor ini yang menjadi nilai pada citra.

(13)	(51)	(52)	(16)	(27)
(28)	(48)	(45)	(3)	(18)
(46)	(46)	(36)	(1)	(49)
(4)	(54)	(19)	(36)	(35)
(31)	(32)	(9)	(47)	(7)

**Gambar VI.** Ilustrasi kombinasi Nilai HSV menjadi satu warna

#### D. Analisis Proses Perhitungan KNN

Penerapan algoritma KNN berperan untuk proses pengklasifikasian citra dengan menghitung *Euclidean Distance* pada tiap fitur dengan membandingkan masing – masing fitur yang dilakukan pada citra uji di setiap pikselnya. Berikut ini merupakan contoh proses perhitungan *Euclidean Distance* pada fitur HSV. Adapun proses perhitungan menghitung jarak kemiripan antara citra latih dan citra uji, dapat dilihat pada Gambar VII dan VIII.

$d_{1i}$	$d_{2i}$	$d_{3i}$	$d_{4i}$	$d_{5i}$
13	51	52	16	27
28	48	45	3	18
46	46	36	1	49
4	54	19	36	35
31	32	9	47	7

$d_{1j}$	$d_{2j}$	$d_{3j}$	$d_{4j}$	$d_{5j}$
59	42	55	21	10
16	11	45	4	2
12	23	36	2	3
24	25	19	9	17
31	44	9	14	27

**Gambar VII**  
vektor hasil Ekstraksi  
HSV citra uji

**Gambar VIII**  
vektor hasil Ekstraksi  
HSV citra latih

$$d(i,j) = \sqrt{|13 - 59|^2 + |51 - 42|^2 + |52 - 55|^2 + \dots + |7 - 27|^2}$$

$$d(i,j) = 109,63119994$$

#### IV. PEMBAHASAN

Proses sistem pendeteksian klasifikasi jenis tanaman daun ini dibangun untuk mengklasifikasi 5 jenis tanaman daun dengan 10 data citra tanaman yang akan diujikan pada klasifikasi tersebut. Pada proses pengujian dari 50 data citra uji yang terdiri dari :

- 10 data citra uji tanaman hias daun *Philodendron Binatifium*,
- 10 data citra uji *Philodendron Keris*,
- 10 data citra uji *Philodendron Black Cardinal*,
- 10 data citra uji *Philodendron Golden Leaf*,
- 10 data citra uji *Aglaonema*.

Pengujian akurasi dilakukan dengan menghitung rata – rata keakuratan sistem pada proses pengklasifikasian di setiap pengujian per-kategori jenis data citra. Adapun data detail pengujian sistem secara keseluruhan, sebagai berikut :

Tabel Detail pengujian sistem secara keseluruhan

Kategori Tanaman hias daun	Jumlah data Uji	Total Benar	Total Salah	Nilai Akurasi
<i>Philodendron Bifinatifium</i>	10	8	2	80%
<i>Philodendron Keris</i>	10	10	0	100%
<i>Philodendron Black Cardinal</i>	10	10	0	100%
<i>Philodendron Golden Leaf</i>	10	8	2	80%
<i>Aglaonema</i>	10	10	0	100%
rata – rata nilai akurasi secara keseluruhan				92%

Sistem Pendeteksi klasifikasi jenis tanaman daun ini beroperasi dengan melakukan ekstraksi fitur warna HSV pada citra disetiap pikselnya, lalu di hitung jarak kemiripan nilai HSV-nya menggunakan metode KNN, setelah didapat nilai yang paling terdekat lalu sistem melakukan perhitungan vote kemiripan yang paling banyak muncul yang akan dijadikan output dari sistem ini.

#### V. KESIMPULAN

Kesimpulan dari penelitian yang telah dilakukan :

1. Klasifikasi jenis tanaman hias daun menggunakan metode *K-Nearest Neighbor* (KNN) berdasarkan nilai Hue, Saturation, Value (HSV) memiliki nilai keakuratan 92% dari data 5 klasifikasi jenis tanaman hias daun yang setiap klasifikasi terdapat

10 data citra yang diujikan, maka total data citra uji yaitu 50 data citra.

2. Dari hasil pengujian didapat 4 data citra uji yang tidak sesuai dengan kategori yang dikarenakan intensitas cahaya citra yang berbeda dengan data latih, untuk solusinya harus ditambahkan proses preprocessing yang dapat mengatur kecerahan pada setiap citra sampel dan menambah fitur tekstur seperti menggunakan metode *Gray Level Co-occurrences Matrices* (GLCM) sehingga hasil yang diperoleh dapat lebih akurat.

Saran yang perlu dipertimbangkan untuk pengembangan penelitian kedepannya, antara lain :

1. Untuk pengembangan sistem klasifikasi ini dapat ditambahkan beberapa ekstraksi fitur seperti ekstraksi tekstur, segmentasi, dan menambahkan ekstraksi fitur warna yang lainnya, sehingga dapat memperoleh hasil yang lebih akurat dan lebih maksimal.
2. Untuk pengembangan dapat ditambahkan algoritma kompresi pada saat proses preprocessing agar proses perhitungan fitur HSV dapat lebih cepat.
3. Proses input dan output dilakukan secara realtime dengan berbasis mobile.

## VI. DAFTAR PUSTAKA

- [1] Prasetyo, E. 2012. *Data Mining – Mengolah Data menjadi Informasi menggunakan Matlab*. Yogyakarta : ANDI.
- [2] Gonzales, R.C., and Woods, R.E. 2002. *Digital Image Processing* Second edition. New Jersey: *Prentice Hall*.
- [3] Agaputra, D. M., Wardani, R. R. K., Siswanto, E., 2013. "Pencarian Citra Digital Berbasis Konten dengan Ekstraksi Fitur HSV, ACD, dan GLCM". Institut Teknologi Harapan Bangsa, Telematika, Vol. 8, No. 2.
- [4] Ch, Dr B Kavitha, Prabhakara Rao, and Dr A. Govardha, "Image retrieval based on color and texture features of the image sub-blocks,," International Journal of Computer Applications, vol. 15, no. 7, 2011.